

UNITED STATES PATENT APPLICATION  
FOR  
MULTIPLE USER INTERFACES FOR AN INTEGRATED FLASH  
DEVICE

INVENTOR:

William R. Stafford

Prepared by:

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP  
12400 WILSHIRE BOULEVARD  
SEVENTH FLOOR  
LOS ANGELES, CA 90025-1026

(408) 720-8598

BSTZ Docket No.: 042390.P6468

Ins.  
A1

# MULTIPLE USER INTERFACES FOR AN INTEGRATED FLASH DEVICE

## FIELD OF THE INVENTION

This invention relates to the field of integrated circuits and, more specifically, to interfaces for integrated circuits.

## 5 BACKGROUND

The motherboard is the main circuit board inside a computer which holds the processor, memory, and expansion slots and connects directly or indirectly to every part of the computer. The motherboard contains a control chipset, a memory device storing code, and the various buses used to link the motherboard components.

Typically, the memory on a motherboard is a Basic Input Output System (BIOS) memory component. BIOS consists of low-level code in a memory component that enables communication between the computer's operating system and hardware, including peripheral technologies. The BIOS also provides the initial instructions for power on self-test and booting system files. The BIOS component is tested by the memory manufacturer to verify compliance with specifications before shipment to the motherboard manufacturer. The motherboard manufacturer then programs the code into the BIOS component. The programming may be performed either before or after the BIOS component is mounted on the motherboard.

A BIOS must also be periodically updated to keep pace with new peripheral technologies. If the BIOS is stored on a read only memory (ROM) chip, it must be replaced. However, if the BIOS is stored on a programmable

memory chip, it may be upgraded using software to reprogram the memory to be compatible with new peripheral technologies.

One prior art BIOS memory component is assembled in a package containing an additional set of pins that are dedicated for use with the programming function performed by the motherboard manufacturer. One problem with such a device is that the use of dedicated pins adds to the total pin count of the packaged device, thereby necessitating the use of a larger package. With the industry trend towards smaller system components, the use of a large package detracts from the desirability of a component.

Furthermore, with prior art BIOS memory components, a device interface specification must be provided to a motherboard manufacturer for programming the memory component. One problem with providing a device interface specification is that it may be undesirable to disclose a proprietary interface to the public.

## SUMMARY OF THE INVENTION

A memory device including a plurality of device circuitry coupled to a plurality of device pads. The memory device also including an operation interface to operate the memory device in an operation mode and a plurality of interfaces to operate the memory device in a plurality of other modes. Each interface of the plurality of interfaces having a predetermined correlation between the plurality of device circuitry and the plurality of device pads. The memory device may also include a configuration circuit to select among the operation interface and the plurality of interfaces.

Additional features and advantages of the present invention will be apparent from the accompanying drawings and from the detailed description that follows.

## BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

5        Figure 1 illustrates one embodiment of a computer system containing a memory component having multiple interfaces.

      Figure 2 illustrates one embodiment of a memory device having multiple interfaces.

      Figure 3 illustrates one embodiment of an interface.

10       Figure 4 illustrates another embodiment of an interface.

      Figure 5 illustrates yet another embodiment of an interface.

      Figure 6 illustrates one embodiment of interface selection circuitry.

## DETAILED DESCRIPTION

A memory device having multiple interfaces is described. The memory device may be configured to operate with different interfaces by using selection circuitry to switch device circuitry connected to device pads.

- 5 For one of the embodiments, the memory device includes a test interface to test the memory device for defects, a programming interface to program the memory device with a code, and an operation interface to operate the memory device in a computer system. The use of multiple interfaces may reduce the pin count of the packaged memory device by providing for the use of existing device pads to program the memory device. A reduced pin count may also allow for integration of non-memory logic functions, for example, general purpose input pins.

Figure 1 illustrates one embodiment of a computer system containing a memory component having multiple interfaces. Computer system 75 includes a processor 60, a memory control hub (MCH) 65, an input/output control hub (ICH) 70, and a memory 50. In one embodiment, memory 50 is a BIOS memory component residing on motherboard 61. In another embodiment, memory 50 may reside on other system boards.

- Sub  
A2  
20 The MCH 65 controls operations between processor 60 and memory devices, for examples, a graphics controller 62 and a random access memory (RAM) 64. The ICH 70 controls operations between processor 60 and input/output (I/O) devices, for examples, a keyboard (KBD) 73 and a mouse 74. The ICH 70 also controls operations between processor 60 and peripheral devices, for example, a drive 71 and a modem 72. In another embodiment, 25 the MCH 65 and the ICH 70 may be integrated into a single component.

Computer system 75 requires an operating system in order to function. The operating system may be stored on a drive of computer system 75, for example, drive 71. In order for the operating system to function, the components on motherboard 61 must be configured to communicate with the  
5 different computer system components. In one embodiment, memory 50 contains firmware that is used for this configuration. In one embodiment, the firmware of memory 50 consists of low-level code that enables communication between the computer's operating system and hardware, for example, modem 72.

10 In one embodiment, memory 50 is a BIOS memory component coupled to a low level bus 66 of computer system 75. The performance of this bus may be low because the firmware is only required to be loaded once, when the computer system 75 is started. When the computer system 75 boots (i.e., is started), a code is fetched from the BIOS memory 50 and placed into the  
15 computer system's 75 random access memory 64. The code is executed by the BIOS to load the operating system of computer system 75 in processor 60.

In one embodiment, the memory 50 is a non-volatile memory, for example, a flash memory, having multiple interfaces that may be selected using configuration circuitry. A nonvolatile memory retains the contents of  
20 data stored within it even after power to the memory is removed. In another embodiment, the device may be other types of programmable memory devices, for examples, PROMs and EPROMs, having multiple interfaces.

In another embodiment, memory 50 is a firmware hub (FWH) component containing multiple interfaces. A firmware hub is a flash  
25 memory component with BIOS functions and security features, containing

dedicated software stored in the memory section of the component. The FWH may be an integrated component of a hub chipset. In an alternative embodiment, memory 50 is not limited to use in personal computer systems, but may also be used in other types of data systems, for example, keyboard  
5 controllers, network hubs, switchers, and workstations.

Figure 2 illustrates one embodiment of a memory device having multiple interfaces. A particular interface may be selected by toggling N configuration lines 253 within BIOS 250, with each interface having M input/output pads 257. In one embodiment, all the interfaces utilize the  
10 same I/O pads on the memory device, thereby reducing the size of the memory device. Each I/O pad may be selectively coupled to different circuitry within the memory device. While one interface is in use, the other interfaces coupled to other device circuitry are disabled.

In one embodiment, BIOS 250 includes two configuration lines 253  
15 selecting from among three interfaces: a test interface 252, a programming interface 254, and an operation interface 256. The use of three interfaces allows for the same device to be accessed through either one of three different modes, for examples, a testing mode, a programming mode, and an operation mode. In another embodiment, BIOS 250 may contain more or less than  
20 three interfaces that may be selected by more or less than two configuration lines. In one embodiment, one of the interfaces is only accessible in the test mode and, thus, is prevented from being accessed in other modes by not connecting one of the configuration lines to an external pin when the memory device is packaged.



The interface I/O pads in the following figures are positioned on two sides of the memory device only for ease of illustration, and the I/O pads connected to particular device circuitry may be positioned anywhere on the physical layout of the device. In one embodiment, the I/O pads are

5 positioned on all four sides of the device. The interfaces may have more or less than the number of device pads illustrated in the figures below.

Sub  
A3

Figure 3 illustrates one embodiment of an interface. In one embodiment, the memory device 350 is configured with a standard flash interface 352 for testing device functionality at wafersort. The standard flash interface 352 includes 20 address pads 301-308, and 313-324; control pads 309, 311-312, and 336-338; configuration pad 339; supply pads 310, and 329-331; data (DQ) pads 325-328, and 332-335; and no connect (NC) pad 340. The control pads 309, 311-312, and 336-339 are used for control operations of the memory device 350, for examples, chip enable, output enable, reset, and status.

10 Configuration pad 339 is used to select between the test interface and the other interfaces used, for example, during the programming and operation modes of memory device 350. During wafersort testing at the component manufacturer, data is loaded into and read from the memory device 350 on data pads 325-328 to test for defects. The 20 pad address configuration allows

15 for accessing the memory device 350 in one cycle because the address information can be loaded in parallel, thereby allowing data to be read on the same cycle.

20

In one embodiment, another one of the multiple interfaces has fewer than 20 address pads, thereby requiring two cycles to perform an addressing

25 operation as discussed in further detail below. As such, use of the test

interface at wafersort test may reduce test time by enabling addressing of the memory device 350 in a single cycle. In addition, a standard flash interface for testing purposes allows for use of existing test hardware and software, thereby maintaining tests costs with minimal added circuitry to the memory device

5 350.

Sub  
A4

Figure 4 illustrates one embodiment of another interface. In one embodiment, interface 454 is a programming interface that may be used in programming the memory device with firmware. As previously discussed, programming is performed after the memory device 350 is assembled into a package and ready for assembly onto the motherboard 61 of Figure 1. During this time, memory device 350 is programmed with low level code to enable the startup of computer system 75 of Figure 1. Configuration pad 402 is bonded out in the packaged memory device 450 for selecting between the programming interface 454 and another interface, for example, a test interface 352 of Figure 3. In one embodiment, configuration pad 439 (used to select the test interface while in test mode) is not connected in the packaged memory device 450 as the test interface is no longer required.

Sub  
A5

Access to the memory device 450 when assembled in a package may be accomplished through the programming interface 454. In one embodiment, the programming interface 454 is no longer used after the computer system 75 is manufactured. In another embodiment, the programming interface may be used at a later time to reprogram the memory device 450 to operate with new technologies. Use of a programming interface 454 having a standard protocol may allow motherboard manufacturers to program the memory device 450

without requiring memory manufacturers to disclose a proprietary interface to motherboard manufacturers.

Sub  
A6

In one embodiment, the programming interface 454 is an Address/Address multiplexer (A/A mux) interface having 11 address pads, 407 and 415-424; control pads 409, 411, 412, and 436-438; configuration pads 402 and 439; supply pads 410 429-431; data pads (DQ) 425-428 and 432-435; and no connect (NC) pads 410, 403-40, 408, 413-414, 440. Data pads 425-428 are used to transfer firmware code data into memory locations (not shown) of memory device 450. The firmware code is programmed into particular memory locations of memory device 450 using address pads 407 and 415-424. Memory data addressing and storage is well known in the art. Accordingly, a detailed description of the memory device's internal components and operation is not provided herein.

In one embodiment, memory locations within memory device 450 are addressed using 22 address locations defining a row and column position of a memory location. As such, in order to load a complete address using an A/A mux interface requires two cycles, as compared with one cycle using a standard flash interface. The low order address (row address: A0-A10) is loaded in the first cycle on the 11 address pads, 407 and 415-424, and the high order address (column address: A11-A21) is loaded in the second cycle using the same 11 address pads, 407 and 415-424.

The use of an interface 454 having half the number of address pads as a standard interface may allow the memory device 450 to be assembled in a smaller pin count package. In one embodiment, memory device 450 is assembled in a 32 pin plastic lead chip carrier (PLCC) package. In another

embodiment, memory device 450 may be assembled in other packages having other pin counts, for example, a 40 pin thin small outline package (TSOP).

Sub  
A7

Although, a second cycle is required to load the memory device with a full address, the A/A Mux interface 454 is suited for use in programming operations where the additional time of using two cycles may not significantly contribute to the total cycle time of programming operations. The memory devices 450 features that are dropped off due to use of a smaller pin count package may include, for examples, locking. Locking is a protection feature that prevents the memory device from being intentionally or inadvertently overwritten. Because the A/A mux interface 454 is usually used only for programming, the additional circuit logic for locking may not be required.

Sub  
A8

Figure 5 illustrates yet another embodiment of an interface. In one embodiment, a third interface 536 is an operation interface. Operation interface 556 is used during operation of memory device 550 in computer system 75 on motherboard 61 of Figure 1. Operation interface 556 includes multiplexed address/data (A/DQ) pads 525-528; configuration pads 502 and 539; supply pads 510, and 529-531; control pads 511-512, 519-520, and 537-538; identification (ID) pads 521-524; clock (CLK) pad 509; and, not connected pads 501, 503-508, 513-518, 532-536, and 540. A/DQ pads 525-528 may be multiplexed to function as both address and data pads. CLK pad 509 may be used to synchronize timing operations of memory device 550 with other chipset components on motherboard 61 of Figure 1.

Sub  
A9

Configuration pad 539 is used to select between the operation interface 556 and the programming interface 456 of Figure 4. The configuration pad 539

is coupled to interface selection circuitry that functions to switch between different memory device circuitry that is coupled to a particular I/O pad. For example, a particular I/O pad connected to control circuitry during operation of the memory device may be switched to address circuitry when

5 programming the device. As such, additional dedicated I/O pads are not needed for programming operations. This eliminates the need for a larger pin count package required when a memory device contains dedicated pads for programming operations.

In one embodiment, the operation interface is a proprietary interface.

10 (The use of a proprietary interface allows for use of a protocol that can be changed at will without requiring the added review time associated with a standard protocol.) In one embodiment, operation interface 556 may not operate in standard flash applications.

Sub A10  
15 A memory device having multiple interfaces enables use of a proprietary interface during device operation while allowing programming of the device through a programming interface that may be released to motherboard manufacturers. Having separate interfaces allows a memory device manufacturer to withhold a device interface specification from tool enabling customers such as BIOS programmers.

26 Sub A11  
25 Figure 6 illustrates one embodiment of interface selection circuitry. Memory device 650 includes selection circuitry coupled to the pads of the device for switching between multiple device interfaces. In one embodiment, the selection circuitry coupled to pad 609, for example, includes control multiplexer 680 and drivers 684, 686, and 688. In one embodiment, drivers 684, 686, and 686 may be coupled to different control function circuitry within

memory device 650. For example, driver 684 may be coupled to chip enable  
circuitry 687 used during testing; driver 686 may be coupled to row/column  
address selection circuitry 689 to toggle between a low and high order address  
during programming, and driver 686 may be coupled to clock circuitry 685  
5 used during in system device operations. Two configuration lines 639 and  
602 are coupled to control multiplexer 680 and used to select between the  
multiple interfaces containing the different device functions.

Drivers and multiplexers are well known in the art. Accordingly, a  
detailed description of their internal components and operation is not  
10 provided herein. The selection circuitry shown in Figure 6 is only for  
illustrative purposes and other selection circuitry configurations may be used  
to select between the multiple device interfaces.

In the foregoing specification, the invention has been described with  
reference to specific exemplary embodiments thereof. It will, however, be  
15 evident that various modifications and changes may be made thereto without  
departing from the broader spirit and scope of the invention as set forth in  
the appended claims. The specification and drawings are, accordingly, to be  
regarded in an illustrative rather than a restrictive sense.